## AMENDMENTS TO THE CLAIMS

1 Please amend Claims 1, 3, 4, 6-14, 16, and 18-21 as follows:

1. (Currently Amended) A method for managing memory used for query execution, the method comprising the computer-implemented steps of:

allocating a buffer map table with locations that correspond to a buffer pool in a cache memory;

wherein the cache memory is managed by a cache memory manager;

wherein the buffer ~~pool~~ map table is managed by a buffer pool manager that is distinct from said cache memory manager;

wherein the buffer pool includes a plurality of buffers;

in response to a need to allocate space in said cache memory for a query working set of a query, the buffer pool manager allocating a buffer from the buffer map table that corresponds to a buffer in the buffer pool to the query working set.

2. (Original) The method of Claim 1 wherein the step of allocating a buffer includes allocating a buffer to store a frame buffer and a bind buffer associated with said query.

3. (Currently Amended) The method of Claim 1 wherein the ~~step of allocating a buffer pool includes allocating a buffer pool in an~~ cache memory is L1 cache that resides on a processor.

4. (Currently Amended) A method as recited in Claim 1, further comprising the step of:

prior to allocating said buffer map table ~~pool~~, determining a size for said buffer pull by calculating a total number of buffers to include in the buffer pool.

5. (Original) A method as recited in Claim 4, wherein the step of calculating a total number of buffers to include in the buffer pool further comprises the steps of:

calculating working set memory requirements of an average query;

sizing buffers in the buffer pool to accommodate the working set memory requirements of the average query;

generating an estimate of how many queries will be executing at a given time; and

setting the number of buffers in the buffer pool to a number sufficient to accommodate the estimated number of queries executed at a given time.

6. (Currently Amended) A method as recited in Claim 4, wherein the step of calculating total number of buffers to include in the buffer pool further comprises the steps of:

calculating working set memory requirements of queries executing during hot sections;

calculating working set memory requirements of frequently used queries;

sizing buffers in the buffer pool to accommodate the working set memory requirements of the queries executing during hot sections and the working set memory requirements of the frequently used queries; and

setting the total number of buffers in the buffer pool to a number sufficient to accommodate queries executed during a hot section.

7. (Currently Amended) A method as recited in Claim 1, wherein the buffer pool manager does not allocate a buffer from the buffer map table ~~pool~~ to a query working set if the working set is larger than the size of buffers within said buffer pool.

8. (Currently Amended) A method as recited in Claim 1, ~~further comprising the step of:~~

~~generating a buffer map table;~~

wherein locations in the buffer map table ~~points to~~ correspond to free and used buffers in the buffer pool; and

wherein the buffer pool manager ~~checks~~ refers to the buffer map table to find ~~for~~ free buffers in the buffer pool prior to allocating buffers in the buffer pool.

9. (Currently Amended) A method as recited in Claim 8, wherein the buffer pool manager allocates a buffer from the buffer pool to a query working set if the buffer map table ~~shows that~~ has a free buffer ~~is free in the buffer pool~~.

10. (Currently Amended) A method as recited in Claim 8, wherein the buffer pool manager does not allocate a buffer from the buffer pool to a query working set if the buffer map table shows that no buffers are free in the buffer pool.

11. (Currently Amended) A method as recited in Claim 8, wherein the buffer pool manager frees a buffer in the buffer pool, without removing the buffer from the buffer pool, after a query releases its allocated buffer by marking a location in the buffer map table that corresponds to the released buffer as free.

12. (Currently Amended) The method of Claim 1 wherein said buffer pool is a first buffer pool, and the method further comprises the steps of:

allocating in ~~said~~ the buffer map table a plurality of locations that correspond to a plurality of buffer pools in the cache memory ~~a plurality of buffer pools~~;

wherein ~~said~~ the plurality of buffer pools include ~~said~~ the first buffer pool and at least one other buffer pool;

wherein each buffer pool of ~~said~~ the plurality of buffer pools is comprised of a plurality of buffers;

wherein each buffer pool of ~~said~~ the plurality of buffer pools has a characteristic that differs from each other buffer pool of ~~said~~ the plurality of buffer pools; and

~~said~~ the buffer pool manager determines which buffer pool of ~~said~~ the plurality of buffer pools should be used to store ~~said~~ the query working set based on the characteristics of ~~said~~ the buffer pools.

13. (Currently Amended) The method of Claim 12 wherein buffers in each buffer pool of the plurality of buffer pools have a different size than buffers in the other buffer pools in ~~said~~ the plurality of buffer pools.

14. (Currently Amended) A method as recited in Claim 12, further comprising, prior to allocating ~~said~~ the plurality of buffer pulls, calculating total number of buffers to include in each buffer pool.

15. (Original) A method as recited in Claim 14, wherein the step of calculating a total number of buffers to include in each buffer pool further comprises the steps of:

calculating working set memory requirements of queries executing during hot sections;

grouping the queries in the hot sections into at least two groups based on common working set memory requirements;

sizing buffers in a particular buffer pool to accommodate the working set memory requirements of queries in one group; and

setting the number of buffers in the particular buffer pool to a number sufficient to accommodate the number of queries executed during a hot section in the one group.

16. (Currently Amended) A method as recited in Claim 15, wherein the total number of buffers in a buffer pool is set so that some queries may be allocated memory from another buffer pool that has buffers that are larger than needed and are unused during a particular hot section.

17. (Original) A method as recited in Claim 12, wherein the buffer pool manager does not allocate a buffer from a buffer pool to a query working set if the query working set is larger than the largest buffer's size.

18. (Currently Amended) A method as recited in Claim 12, ~~further comprising the step of:~~

~~providing a buffer map table;~~

wherein locations in the buffer map table ~~points to~~ correspond to free and used buffers in the buffer pool; and wherein the buffer pool manager ~~checks~~ refers to the buffer map table to find ~~for~~ free buffers in the buffer pool prior to allocating buffers in the buffer pool.

19. (Currently Amended) A method as recited in Claim 18, wherein the buffer pool manager allocates a buffer from the buffer pool to a query working set if the buffer map table ~~shows that~~ has a free buffer ~~is free in the buffer pool~~.

20. (Currently Amended) A method as recited in Claim 18, wherein the buffer pool manager does not allocate a buffer from the buffer pool to a query working set if the buffer map table shows that no buffers are free in the buffer pool.

21. (Currently Amended) A method as recited in Claim 18, wherein the buffer pool manager frees a buffer in the buffer pool after a query releases its allocated buffer by marking a location in the buffer map table that corresponds to the released buffer as free.

22. (Original) A computer-readable medium carrying one or more sequences of instructions which, when executed by one or more processors, causes the one or more processors to perform the method recited in Claim 1.

23. (Original) A computer-readable medium carrying one or more sequences of instructions which, when executed by one or more processors, causes the one or more processors to perform the method recited in Claim 2.

24. (Original) A computer-readable medium carrying one or more sequences of instructions which, when executed by one or more processors, causes the one or more processors to perform the method recited in Claim 3.

25. (Original) A computer-readable medium carrying one or more sequences of instructions which, when executed by one or more processors, causes the one or more processors to perform the method recited in Claim 4.

26. (Original) A computer-readable medium carrying one or more sequences of instructions which, when executed by one or more processors, causes the one or more processors to perform the method recited in Claim 5.

27. (Original) A computer-readable medium carrying one or more sequences of instructions which, when executed by one or more processors, causes the one or more processors to perform the method recited in Claim 6.

28. (Original) A computer-readable medium carrying one or more sequences of instructions which, when executed by one or more processors, causes the one or more processors to perform the method recited in Claim 7.

29. (Original) A computer-readable medium carrying one or more sequences of instructions which, when executed by one or more processors, causes the one or more processors to perform the method recited in Claim 8.

30. (Original) A computer-readable medium carrying one or more sequences of instructions which, when executed by one or more processors, causes the one or more processors to perform the method recited in Claim 9.

31. (Original) A computer-readable medium carrying one or more sequences of instructions which, when executed by one or more processors, causes the one or more processors to perform the method recited in Claim 10.

32. (Original) A computer-readable medium carrying one or more sequences of instructions which, when executed by one or more processors, causes the one or more processors to perform the method recited in Claim 11.

33. (Original) A computer-readable medium carrying one or more sequences of instructions which, when executed by one or more processors, causes the one or more processors to perform the method recited in Claim 12.

34. (Original) A computer-readable medium carrying one or more sequences of instructions which, when executed by one or more processors, causes the one or more processors to perform the method recited in Claim 13.

35. (Original) A computer-readable medium carrying one or more sequences of instructions which, when executed by one or more processors, causes the one or more processors to perform the method recited in Claim 14.

36. (Original) A computer-readable medium carrying one or more sequences of instructions which, when executed by one or more processors, causes the one or more processors to perform the method recited in Claim 15.

37. (Original) A computer-readable medium carrying one or more sequences of instructions which, when executed by one or more processors, causes the one or more processors to perform the method recited in Claim 16.

38. (Original) A computer-readable medium carrying one or more sequences of instructions which, when executed by one or more processors, causes the one or more processors to perform the method recited in Claim 17.

39. (Original) A computer-readable medium carrying one or more sequences of instructions which, when executed by one or more processors, causes the one or more processors to perform the method recited in Claim 18.

40. (Original) A computer-readable medium carrying one or more sequences of instructions which, when executed by one or more processors, causes the one or more processors to perform the method recited in Claim 19.

41. (Original) A computer-readable medium carrying one or more sequences of instructions which, when executed by one or more processors, causes the one or more processors to perform the method recited in Claim 20.

42. (Original) A computer-readable medium carrying one or more sequences of instructions which, when executed by one or more processors, causes the one or more processors to perform the method recited in Claim 21.